Access control in IoT environments: Feasible scenarios

Yasmina Andaloussi, Driss El Ouadghiri, Yoann Maurel, Jean-Marie Bonnin, Habiba Chaoui

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 130 (2018) 1031–1036

Procedia Computer Science

www.elsevier.com/locate/procedia

The 8th International Symposium on Frontiers in Ambient and Mobile Systems (FAMS-2018)

Access control in IoT environments: Feasible scenarios

Y. Andaloussi[1]*, M. D. El Ouadghiri[1], Y. Maurel[2], J. M. Bonnin[3], H. Chaoui[4]

[1]*IA Laboratory, Science Faculty, My Ismail University, Meknes, Morocco*
[2]*ISTIC, Université de Rennes 1 - IRISA, Equipe TACOMA, Rennes, France*
[3]*Institut Mines Telecom Atlantique - IRISA, Equipe TACOMA, Rennes, France*
[4]*ENSA, Ibn-Tofail University, Kenitra, Morocco*

Abstract

The Internet of Things (IoT) is the extension of the internet to the physical world where all objects collect information and interact with their environments with no or little human intervention. They collect and transfer sensitive and private data from various users. This puts security and privacy issues at the forefront: the ability to manage the digital identity of millions of people and billions of devices is fundamental for success. As most of the information contained in IoT environment may be personal or sensitive data, there is a requirement to support anonymity and restrain access to information. This article will focus on access control and authentication mechanisms as well as supporting the cryptography algorithms in constrained devices.

Peer-review under responsibility of the Conference Program Chairs.

Keywords: IoT; access control; cryptography; security; smart objects; DCapBAC

1. Introduction

In recent years, advances in the fields internet technologies have given rise to new possibilities and visions, which envisage the world as a vast network of autonomous, self-organized and self-configuring devices. This vision has

* Corresponding author. Tel.: +212 6 65 88 06 47; fax: +212 5 35 45 43 01.
E-mail address: andyasmina@gmail.com

1877-0509 © 2018 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the Conference Program Chairs.
10.1016/j.procs.2018.04.144

produced a paradigm called the Internet of Things (IoT). Thus, access control has become a real challenge in the IoT.

Access control is a system that allows an authority to control access to zones and resources of a given installation. It ensures confidentiality in such a way as to ensure that information is only accessible to those authorized, it assumes also integrity in such a way that the data are indeed those believed to be. Most of recent proposals have addressed the problem of access control using centralized approaches where a central entity is responsible for managing the authorization mechanisms, allowing or denying requests from external entities[6]. Although in these approaches, end-to-end security between devices and any Internet host cannot be achieved. However, traditional access control models do not meet the requirements imposed by IoT scenarios, introducing lack of flexibility, scalability and usability in environments with billions of devices. These problems could be solved by a distributed approach in which "things" are able to make authorization decisions without delegating this process to a different entity.

In this paper, we will be interested on the capability-based access control model. This model grants some interesting aspects such as the principle of least privilege, and greater adaptation to the requirements of IoT scenarios. The rest of this paper is structured as follows: Section 2 describes related works regarding access control in IoT, Section 3 gives a description of the different architectures of access control mechanisms. Section 4 itemizes the capability-based access control. Section 5 shows a detailed overview of our proposal. And section 6 end up with some conclusions and future works in this area.

2. Backgrounds

Nowadays, there is a myriad of access control models that are applied to different Internet of Things scenarios in which security is required. In the following, we give a brief description of the most popular models, which are deployed in such scenarios.

In the Mandatory Access Control (MAC) model[1] the administrator of the system give permissions for subject to access object. The model assigns security labels to subjects and objects, and it is independent of the user operations, only the administrator can modify object security labels. MAC models are difficult and expensive to implement and maintain, its usage is usually limited to military applications, and this is why MAC models are not used as access control system.

In the Discretionary Access Control (DAC) models[2], the access to resources is maintained by users, which can grant permissions to their resources by being included in Access Control Lists (ACL). Each entry in the access control list gives users (or group of subjects) permissions to access resources. The permissions are usually stored by objects. Unlike in MAC, where permissions are given in predefined policies by the administrator, in DAC, permissions are given by users which decide the access rights to the resources they belong. DAC is adopted by current operation systems based on UNIX, FreeBSD, and Windows.

Moreover, in the Role-Based Access Control (RBAC) model[3], users are assigned to roles, and the security policies grant rights to roles rather than to users. Since the users are associated to roles. RBAC allows creating hierarchies of permissions and inheritance. Nonetheless, RBAC has some problems since the administrative issues of large systems where memberships make administration potentially cumbersome.

Traditional access control models like MAC, DAC and RBAC do not take into account additional parameters such as resource information and dynamic information (such as time, location …). In order to provide a more flexible mechanism, the Attribute-Based Access Control (ABAC) model[4] was proposed, in which authorization decisions are based on attributes that the user has to prove (e.g.: age, location, roles, etc.). One of the main advantages of ABAC is requesters do not have to be known a priori by targets, providing a higher level of flexibility for open

environments, compared to RBAC models. Nevertheless, in ABAC everyone must agree on a set of attributes and their meaning when using ABAC, which is not easy to accomplish.

The Authorization-Based Access Control (ZBAC)[5] model uses authorization credentials, which are presented along a request to make an access control decision. Unlike ABAC and RBAC systems, in which the user submits an authentication along with the service request, in ZBAC systems, the user submits an authorization along with the request.

IoT scenarios imposes significant restrictions on privacy and access control, tradition access control approaches solutions were not designed with these aspects.

3. IoT Access Control Architectures

In this section we present the approaches related to access control, their main advantages and drawbacks.

3.1. Centralized approach

In a centralized approach[6,7] , all access control logic is externalized into a central entity responsible for filtering access requests based on their authorization policies. The end devices (sensors …) play a limited role as information providers. This centralized approach does not take into consideration constraints of resources, because the access control logic is located in an entity without constraints of resources.

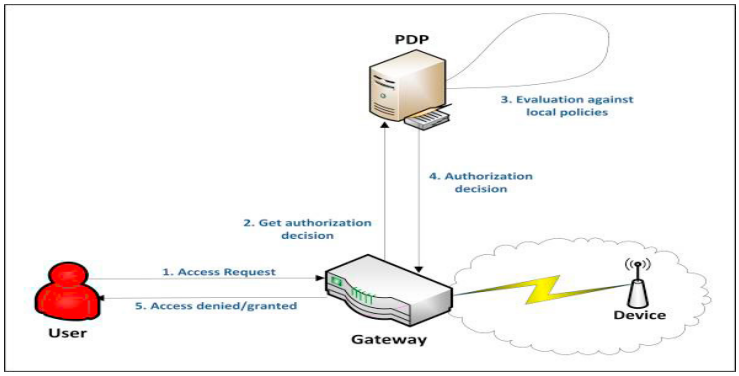

Fig. 1. Centralized approach[6].

This approach is expressed by a scenario where a server receives a request from the mobile user who wants to access the end-device, so it generates a token containing the authorization or the refusal and sends it to the mobile user. However, this approach has major problems. Firstly, the end-device is not taken into consideration in access control decisions. Secondly, the access control logic is located in one entity, so any vulnerability might compromise all the system because it becomes a single point of failure.

3.2. Hybrid approach

The end-devices are not passive entities[6,8], they participate in the access control decisions such as presented in the Figure 2. In this approach, a server receives a request from the mobile user who wants to access the end-device, it evaluates the contextual information provided by the end device and then decides to allow or reject the access, so it generates a token containing the authorization or the refusal and sends it to the mobile user. This approach has also problem in the ability of the end-devices to take contextual information into account (location, time …) in the exact time for making decisions. So the information loses the importance in the process of authorization decision.

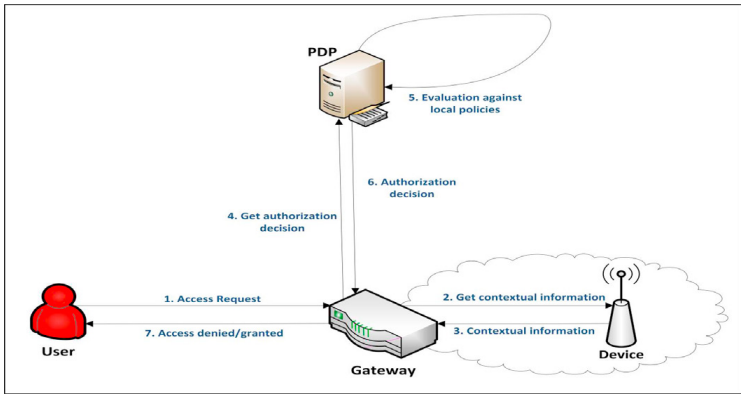

Fig. 2. Hybrid approach[6].

3.3. Distributed approach

In the distributed architectures[6], the end-device is a smart thing that is enable to obtain process and send information to other services and devices. The devices are able to take authorizations decisions without the need of central entities.

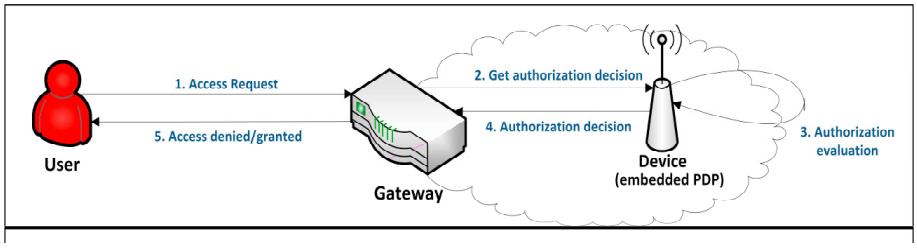

Fig. 3. Distributed approach[6].

This approach presents interesting features and it is more suitable for the scenarios and architectures of IoT.

4. Design of capability-based access control

4.1. Capability-based access control

DCapBAC has been postulated as a feasible approach to be deployed on IoT scenarios even in the presence on devices with resource constraints. The key concept of this approach is the concept of capability, which was originally introduced by[9] as "token, ticket, or key that gives the possessor permission to access an entity or object in a computer system". This token is usually composed by a set of privileges which are granted to the entity holding the token. In CapBAC[10,11], an entity, which wants to access certain information from another entity, requires to send a token together the request. Thus, the entity that receives the capability already knows the permissions that the requester has been granted when need to process the request. This simplifies the authorization mechanism and it is a relevant feature in scenarios with resources-constrained devices and complex access control policies are not necessary.

4.2. Capability token

We are building an implementation of capability-based access control; the capability token is represented by

JSON[12] for its suitability in constrained devices. A brief description of each field is provided in[12].

5. Our approach

5.1. Our architecture

A test bed was implemented to demonstrate the feasibility of the approach. The elements of our architecture are: Raspberry PI, the issuer, responsible for generating the capability token.ESP 8266, the end-device that will be read the request, verify the signature and decide to accept or not the request. The resource-constrained device of our scenario was implemented in an ESP 8266, with a 64 KiB of instruction RAM, 96 KiB of data RAM. Moreover, the Raspberry PI has been implemented in Java and the ESP 8266 has been implemented in C, C++. The capability token that we used is not like the capability token in the literature because it grants integrity and legitimacy of each device, each device is able to decrypt and verify the signature only of the part intended for him. So it can have access only to the part of the capability token proper to him. We used Elliptic Curve Cryptography (ECC) and Elliptic Curve Digital Signature Algorithm (ECDSA) to encrypt and sign the token because of its suitability in constrained devices.

Table 1. Comparaison entre ECC et RSA (paramètres de NIST).

RSA Key (Bits)	ECC Key (Bits)
1024	160
2048	224
3072	256
7680	384

5.2. Process of decision

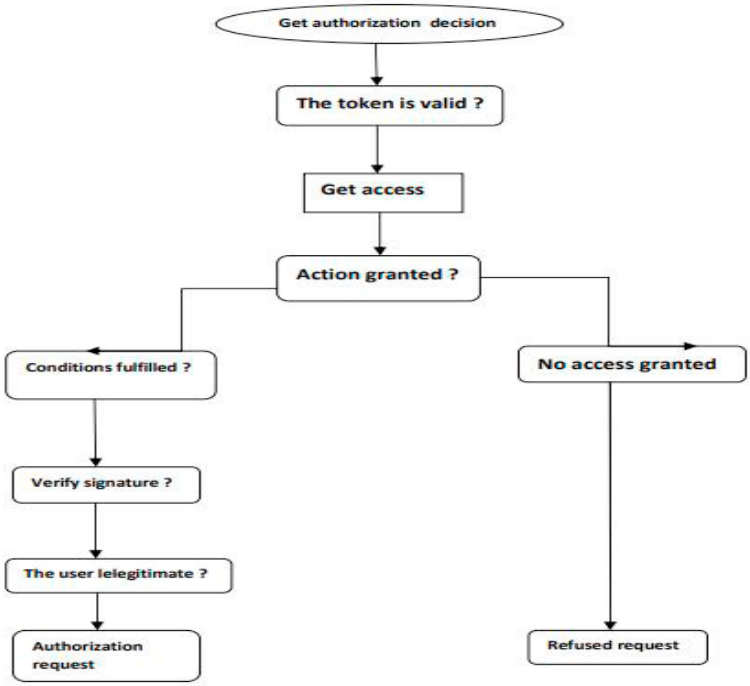

Fig. 4. Authorization scenario.

- **Check the validity of the token:** The device verify the validity of the fields NB, NA and II, if the token is not valid, the request is refused.
- **Check if the action is permitted:** The device verify each one of the access rights AC.
- **Check that the actions are fulfilled:** The device verifies the permissions of each resource.
- **Check the validity of the signature:** The device verifies the signature of the token, so the device will recalculate the signature and compare it with the signature of the token.
- **Check the legitimacy of the user:** if all the actions are granted, the access is permitted.

6. Conclusion

This article covered the CapBAC model that was proposed as a feasible approach for IoT scenarios[7] and is supported by constrained devices. This approach is based on the concept of capability which is defined in[6,8]. This token contains the privileges that will be granted to the entity that holds the token. This token must be tamper-proof and identified unequivocally to be considered in a real environment. Moreover, it is necessary to have a cryptographic mechanism which is supported even by the constrained devices.

The security is a challenge in IoT environments and new technologies. This implementation is taken into account the constraints of smart objects in term of storage, energy consumption and execution time.

References

1. R. S. (1993). *Lattice-based access control models*. Computer, 26(11), 9-19.
2. Moffett, J., Sloman, M., & Twidle, K. (1990). *Specifying discretionary access control policy for distributed systems*. Computer Communications, 13(9), 571-580.
3. Sandhu, R. S., Coyne, E. J., Feinstein, H. L., & Youman, C. E. (1996). *Role-based access control models*. Computer, 29(2), 38-47.
4. Vincent C. et a. NIST Special Publication 800-162. *Guide to Attribute Based Access Control (ABAC) Definition and Considerations* http://nvlpubs.nist.gov/nistpubs/specialpublications/NIST.sp.800-162.pdf
5. From ABAC to ZBAC: The Evolution of Access Control Models http://www.hpl.hp.com/techreports/2009/HPL-2009-30.pdf
6. L. Hernandez-Ramos, J. Jara, L.Martin, F. Skarmeta, *Distributed Capability-based Access Control for the Internet of Things*. Journal of Internet Services and Information Security (JISIS), volume: 3, number: 3/4, pp. 1-16
7. R. Rom´an, J. Zhou, and J. L´opez. *On the Features and Challenges of Security and Privacy in Distributed Internet of Things*. Computer Networks, 57(10):2266–2279, July 2013.
8. Z.Shelby,K.Hartke,andC.Bormann. *Constrained Application Protocol(CoAP)*. IETFInternet-draft(work in progress), June 2013. http://tools.ietf.org/html/draft-ietf-core-coap-18.
9. J. Dennis and E. V. Horn. *Programming Semantics for Multiprogrammed Computations*. Communications of the ACM, 9(3):143–155, 1966.
10. C.Ellison, B.Frantz, B.Lampson, R.Rivest, B.Thomas,andT.Ylonen. RFC2693: SPKICertificateTheory. IETF RFC 2693, September 1999. http://www.ietf.org/rfc/rfc2693.txt.
11. S. Gusmeroli, S. Piccione, and D. Rotondi. *A capability-based security approach to manage access control in the internet of things*. Mathematical and Computer Modelling, 58(5-6):1189–1205, September 2013.
12. D.Ferraiolo,J.Cugini, andR.Kuhn. *Role-based access control (RBAC):Features and motivations*. InProc. of 11th Annual Computer Security Application Conference, pages 241–248, 1995.